

FIG. 1

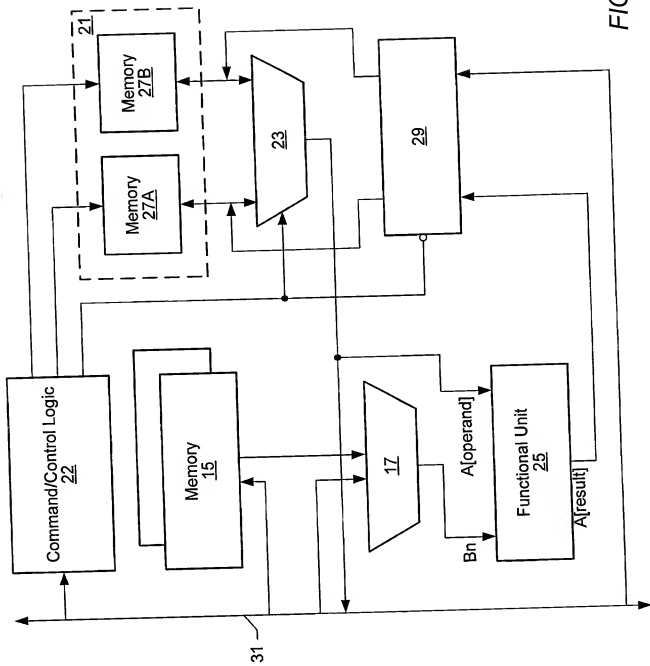
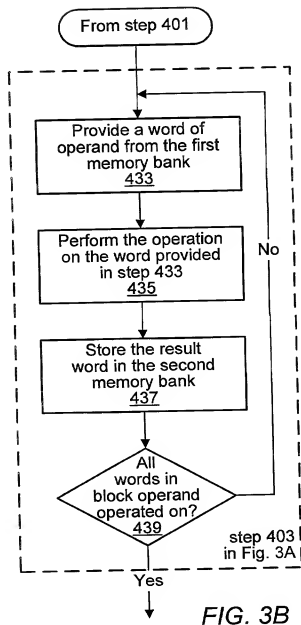
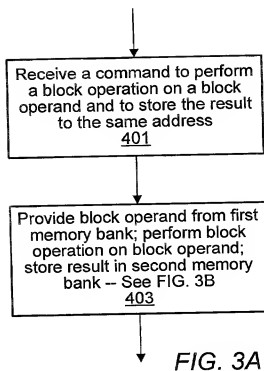


FIG. 2



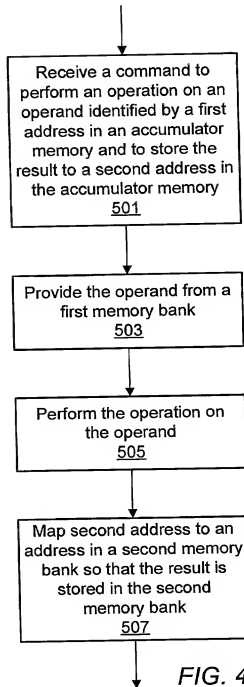


FIG. 4

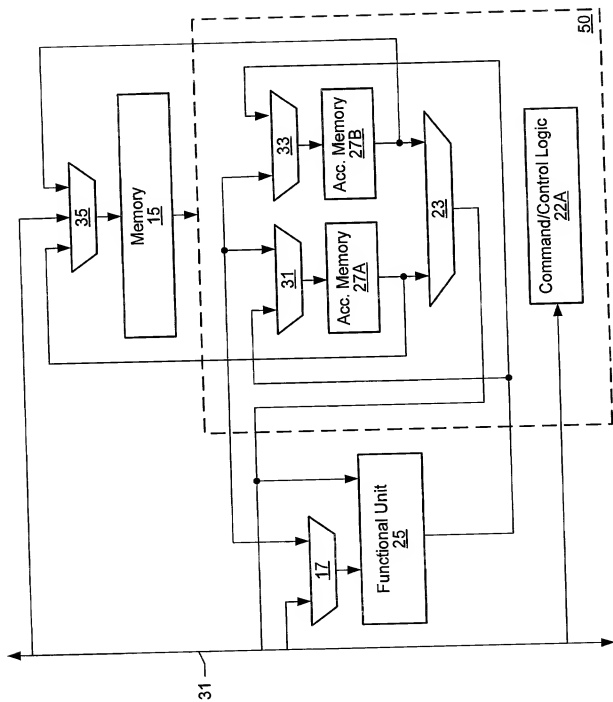


FIG. 5

Instructions	Acc. Memory 27A	Acc. Memory 27B	Memory 15
$B = B0$	$B(\text{new})$	n/a	B0
$B = B \text{ XOR } B1$	$B(\text{old})$	$B(\text{new})$	B1
$B = B \text{ XOR } B2$	$B(\text{new})$	$B(\text{old})$	B2
$B = B \text{ XOR } B3$	$B(\text{old})$	$B(\text{new})$	B3
$B = B \text{ XOR } B4$	$B(\text{new})$	$B(\text{old})$	B4
flush cache	n/a	n/a	B

FIG. 6

Instructions	Acc. Memory <u>27A</u>	Acc. Memory <u>27B</u>	Memory <u>15</u>
B = immed(B0)	B(new)	n/a	n/a
B = B XOR immed(B1)	B(old)	B(new)	n/a
B = B XOR immed(B2)	B(new)	B(old)	n/a
B = B XOR immed(B3)	B(old)	B(new)	n/a
B = B XOR immed(B4)	B(new)	B(old)	n/a
flush cache	n/a	n/a	B

FIG. 7

Accumulation Operations:

$B = B0 \text{ XOR } B1 \text{ XOR } B2 \text{ XOR } B3 \text{ XOR } B4$

$C = C0 \text{ XOR } C1 \text{ XOR } C2 \text{ XOR } C3 \text{ XOR } C4$

$D = D0 \text{ XOR } D1 \text{ XOR } D2 \text{ XOR } D3 \text{ XOR } D4$

Instructions:

- (1) $B = B0$
- (2) $B = B \text{ XOR } B1$
- (3) $B = B \text{ XOR } B2$
- (4) $C = C0$
- (5) $C = C \text{ XOR } C1$
- (6) $C = C \text{ XOR } C2$
- (7) $B = B \text{ XOR } B3$
- (8) $D = D0$
- (9) $D = D \text{ XOR } D1$
- (10) $D = D \text{ XOR } D2$
- (11) $C = C \text{ XOR } C3$
- (12) $D = D \text{ XOR } D3$
- (13) $B = B \text{ XOR } B4$
- (14) $C = C \text{ XOR } C4$
- (15) $D = D \text{ XOR } D4$

FIG. 8A

Row#	Instructions/Operations	Acc. Memory <u>27A</u>	Acc. Memory <u>27B</u>	Memory <u>15</u>
1	(1) B = B0	B(new)	n/a	B0
2	(2) B = B XOR B1	B(old)	B(new)	B1
3	(3) B = B XOR B2	B(new)	B(old)	B2
4	(4) C = C0	C(new)	n/a	C0
5	(5) C = C XOR C1	C(old)	C(new)	C1
6	(6) C = C XOR C2	C(new)	C(old)	C2
7	(7) B = B XOR B3	B(old)	B(new)	B3
8	(8) D = D0	n/a	n/a	n/a
9	(8) flush C	C	n/a	C
10	(8) load D	D(new)	n/a	D0
11	(9) D = D XOR D1	D(old)	D(new)	D1
12	(10) D = D XOR D2	D(new)	D(old)	D2
13	(11) C = C XOR C3	n/a	n/a	n/a
14	(11) flush B	n/a	B	B
15	(11) load C	n/a	C(new)	C
16	(11 completes)	C(new)	C(old)	C3
17	(12) D = D XOR D3	D(old)	D(new)	D3
18	(13) B = B XOR B4	n/a	n/a	n/a
19	(13) flush C	C	n/a	C
20	(13) load B	B(new)	n/a	B
21	(13 completes)	B(old)	B(new)	B4
22	flush B	n/a	B	B
23	(14) C = C XOR C4	n/a	n/a	n/a
24	(14) load C	n/a	C(new)	C
25	(14 completes)	C(new)	C(old)	C4
26	flush C	C	n/a	C
27	(15) D = D XOR D4	D(new)	D(old)	D4
28	flush cache	D	n/a	D

FIG. 8B

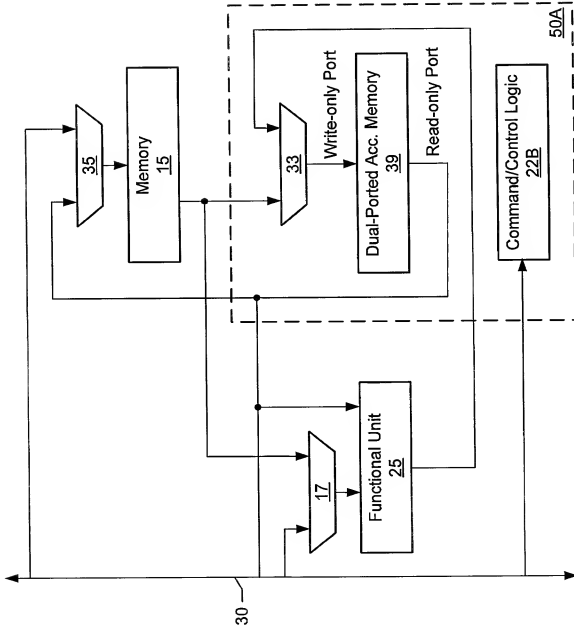


FIG. 9

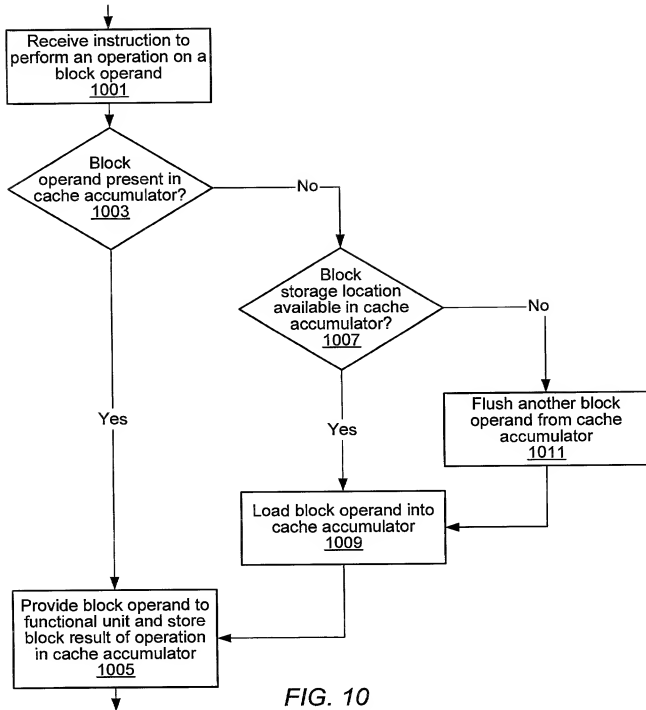
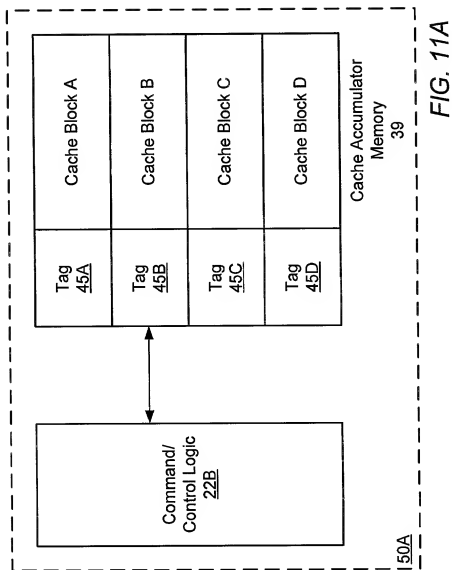
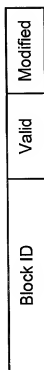


FIG. 10



Tag 45A



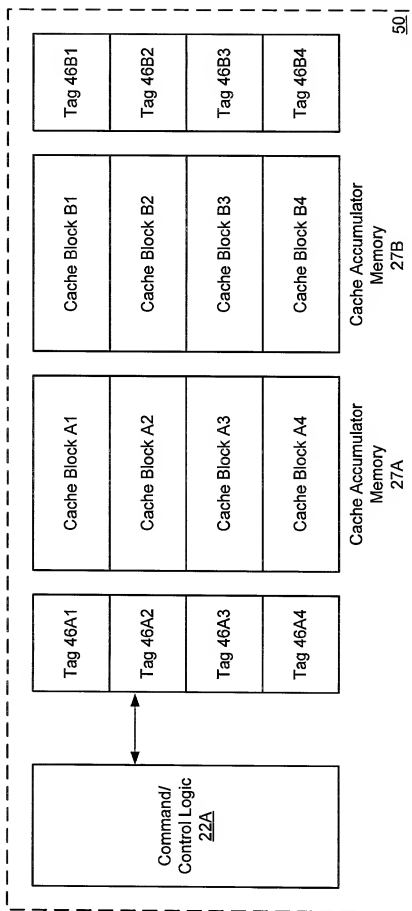


FIG. 12A

Tag
46A1

Block ID	Valid	Modified	Bank
----------	-------	----------	------

FIG. 12B

Accumulation Operation:
 $D = A \text{ XOR } B \text{ XOR } C$

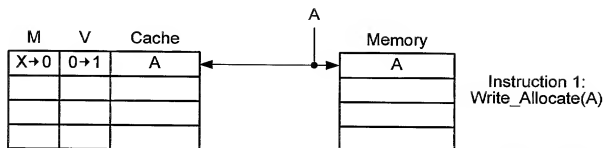


FIG. 13A

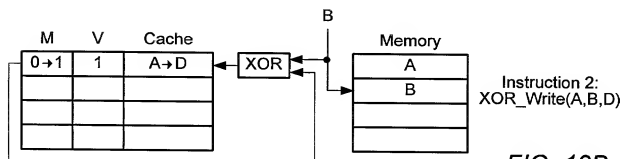


FIG. 13B

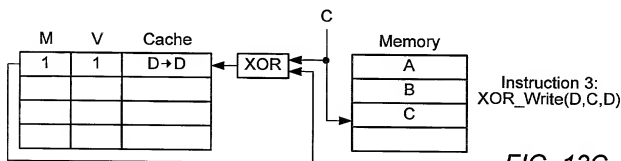


FIG. 13C

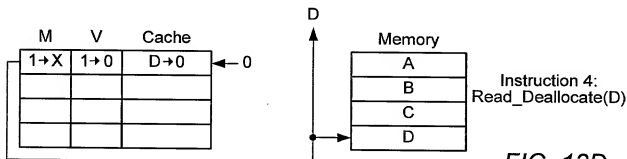
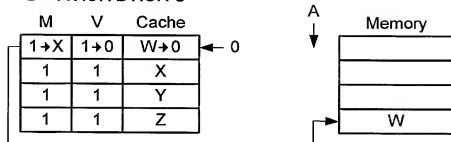


FIG. 13D

Accumulation Operation:

$$D = A \text{ XOR } B \text{ XOR } C$$



Instruction 1:
Write_Allocate(A)

All block storage locations
are allocated; W is flushed
to memory

FIG. 14A

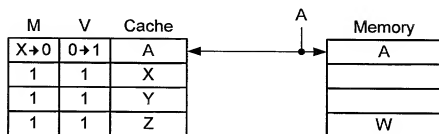
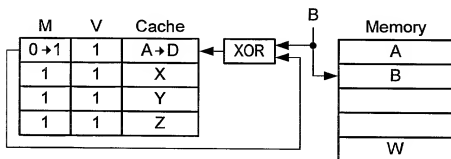
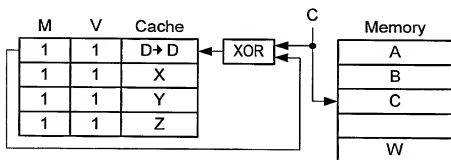


FIG. 14B



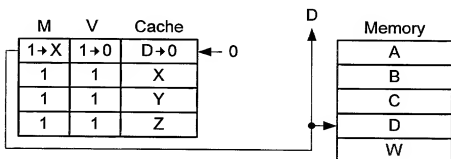
Instruction 2:
XOR_Write(A,B,D)

FIG. 14C



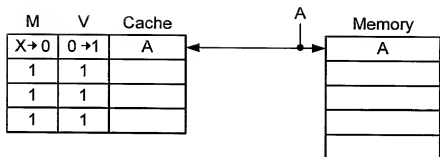
Instruction 3:
XOR_Write(D,C,D)

FIG. 14D



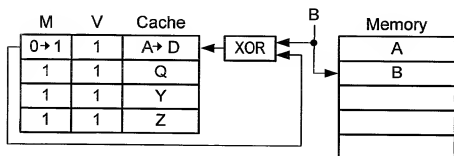
Instruction 4:
Read_Deallocate(D)

FIG. 14E



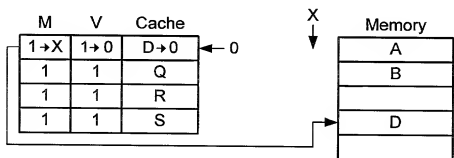
Instruction 1:
Write_Allocate(A)

FIG. 15A



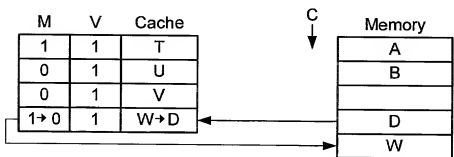
Instruction 2:
XOR_Write(A,B,D)

FIG. 15B



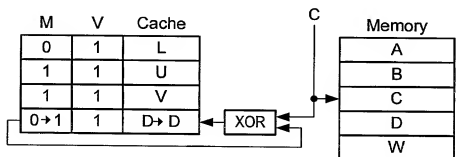
D is flushed to
memory.

FIG. 15C



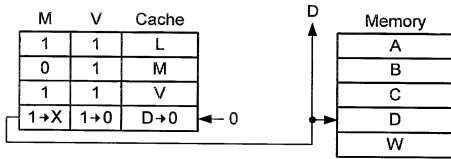
Instruction 3:
XOR_Write(D,C,D)

FIG. 15D



After D has been loaded into
the cache accumulator,
instruction 3 is executed.

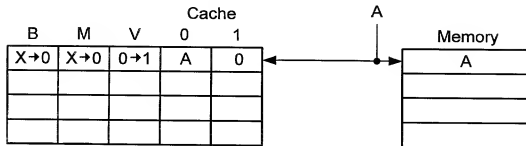
FIG. 15E



Instruction 4:
Read_Deallocate(D)

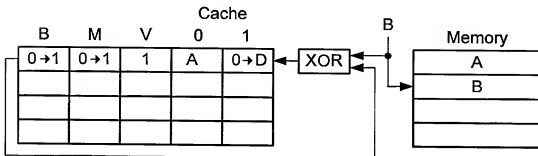
FIG. 15F

Accumulation Operation:
 $D = A \text{ XOR } B \text{ XOR } C$



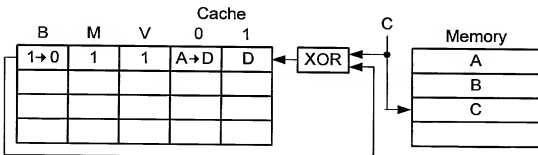
Instruction 1:
Write_Allocate(A)

FIG. 16A



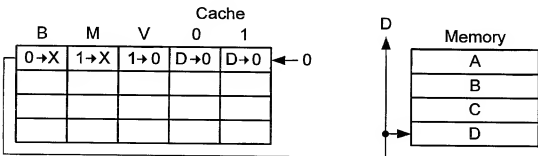
Instruction 2:
XOR_Write(A,B,D)

FIG. 16B



Instruction 3:
XOR_Write(D,C,D)

FIG. 16C



Instruction 4:
Read_Deallocate(D)

FIG. 16D

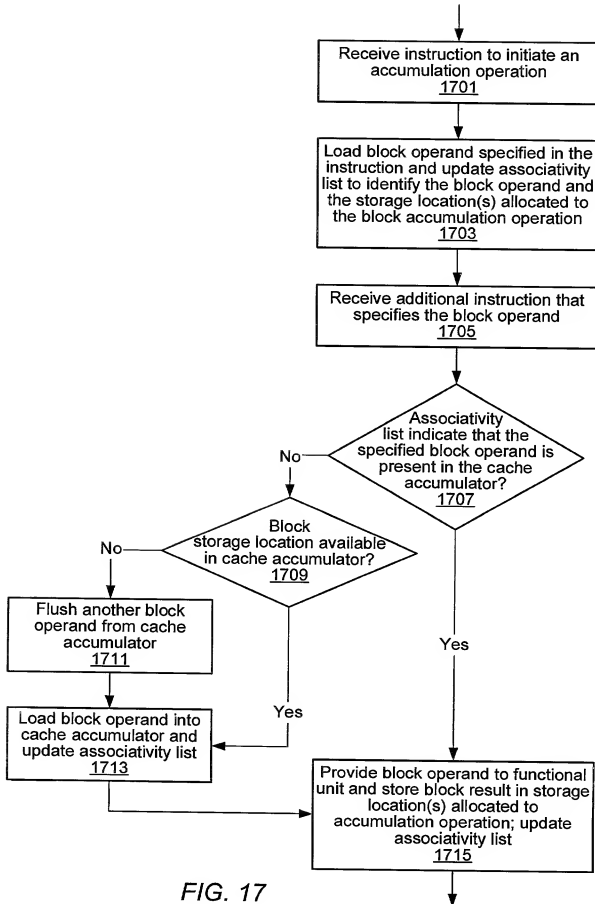


FIG. 17